JP919990195
H. TAI, ET AL

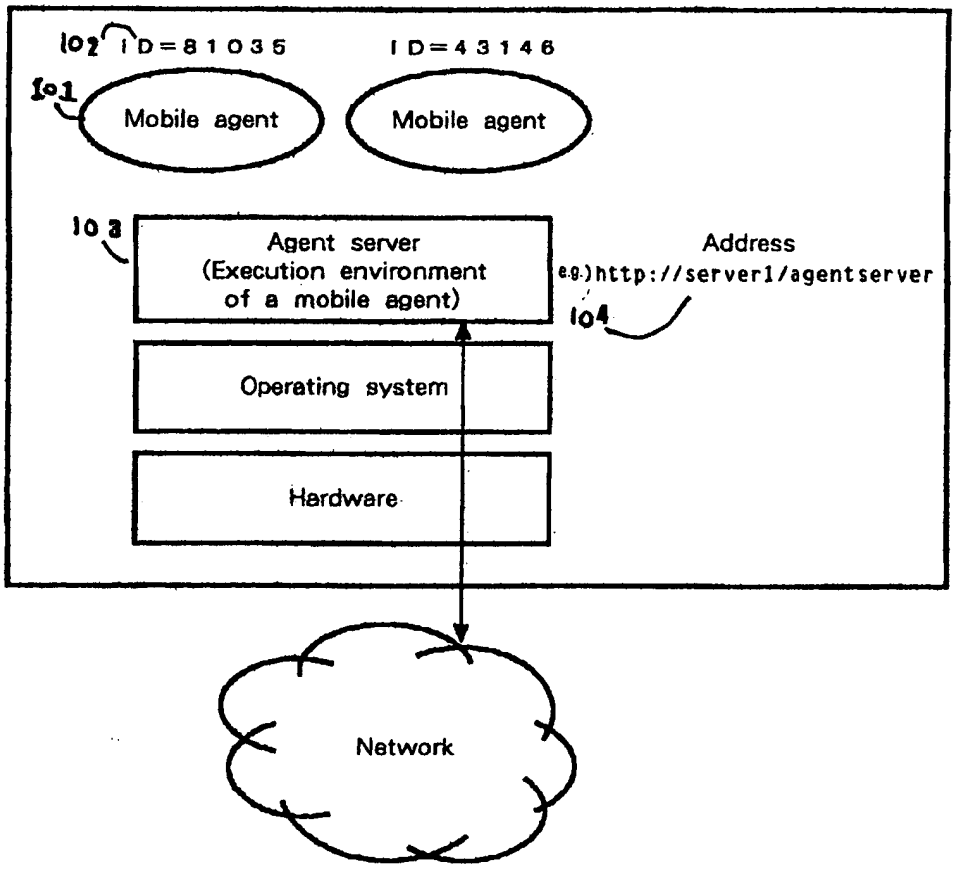

Fig. 1

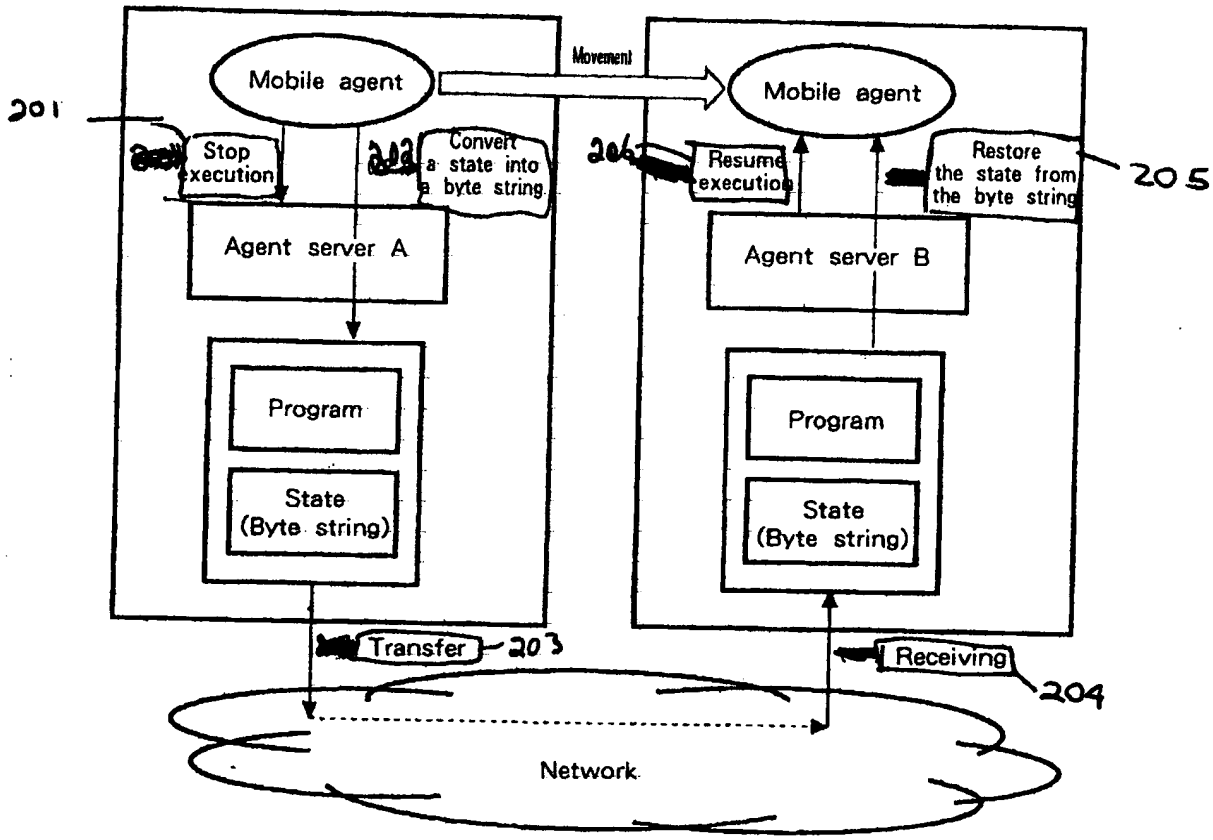
Mobile agent
Movement
Mobile agent
201
Stop execution
Convert a state into a byte string
Resume execution
Restore the state from the byte string
205
Agent server A
Agent server B
Program
State (Byte string)
Program
State (Byte string)
Transfer
203
Receiving
204
Network

Fig. 2

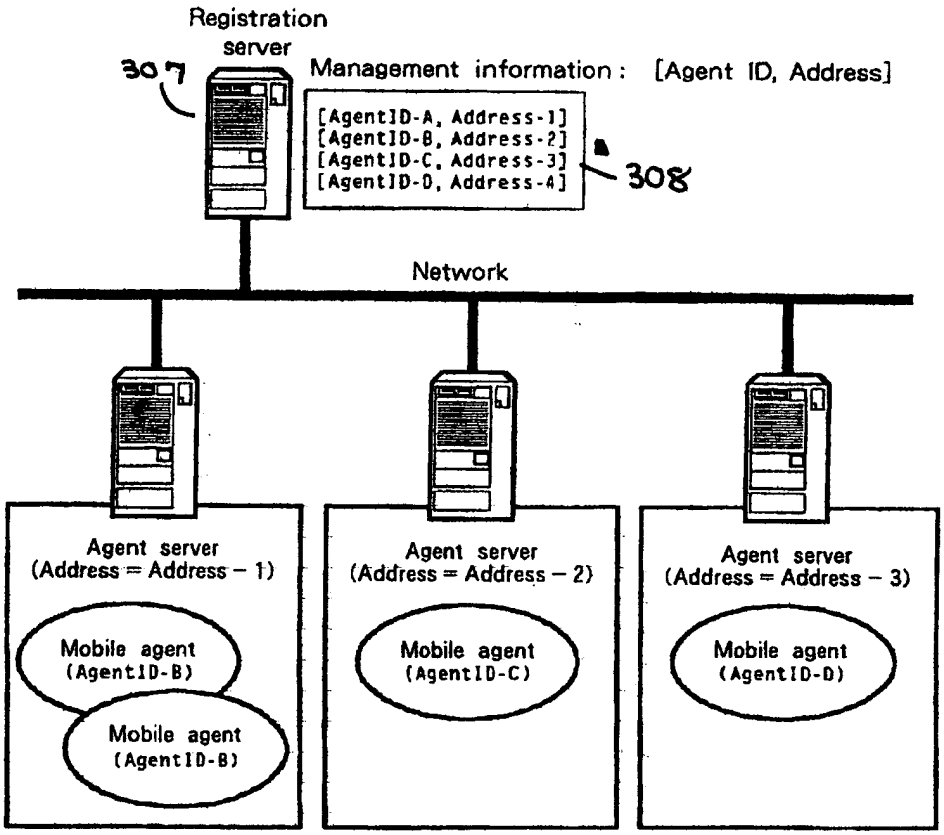
Registration server
307
Management information : [Agent ID, Address]
[AgentID-A, Address-1]
[AgentID-B, Address-2]
[AgentID-C, Address-3]
[AgentID-D, Address-4]
308
Network
Agent server (Address = Address – 1)
Mobile agent (AgentID-B)
Mobile agent (AgentID-B)
Agent server (Address = Address – 2)
Mobile agent (AgentID-C)
Agent server (Address = Address – 3)
Mobile agent (AgentID-D)

Fig. 3

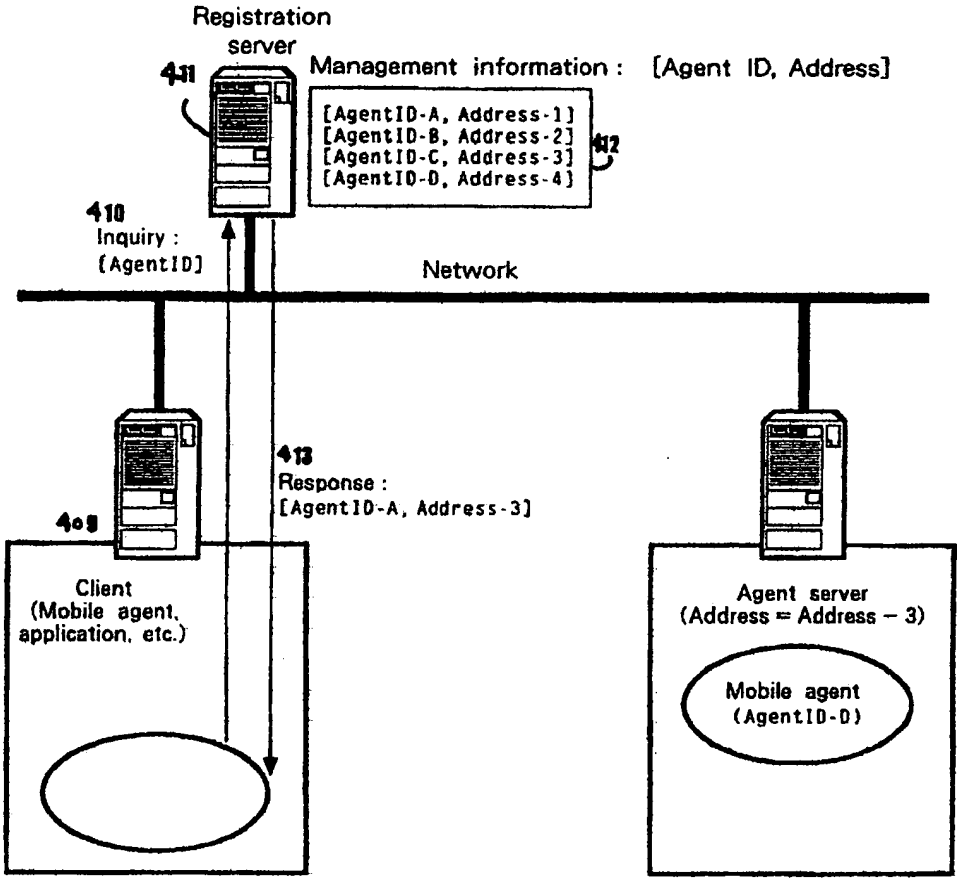
Registration server
411
Management information : [Agent ID, Address]
[AgentID-A, Address-1]
[AgentID-B, Address-2]
[AgentID-C, Address-3]
[AgentID-D, Address-4]
412
410
Inquiry :
[AgentID]
Network
413
Response :
[AgentID-A, Address-3]
408
Client
(Mobile agent, application, etc.)
Agent server
(Address = Address – 3)
Mobile agent
(AgentID-D)

Fig. 4

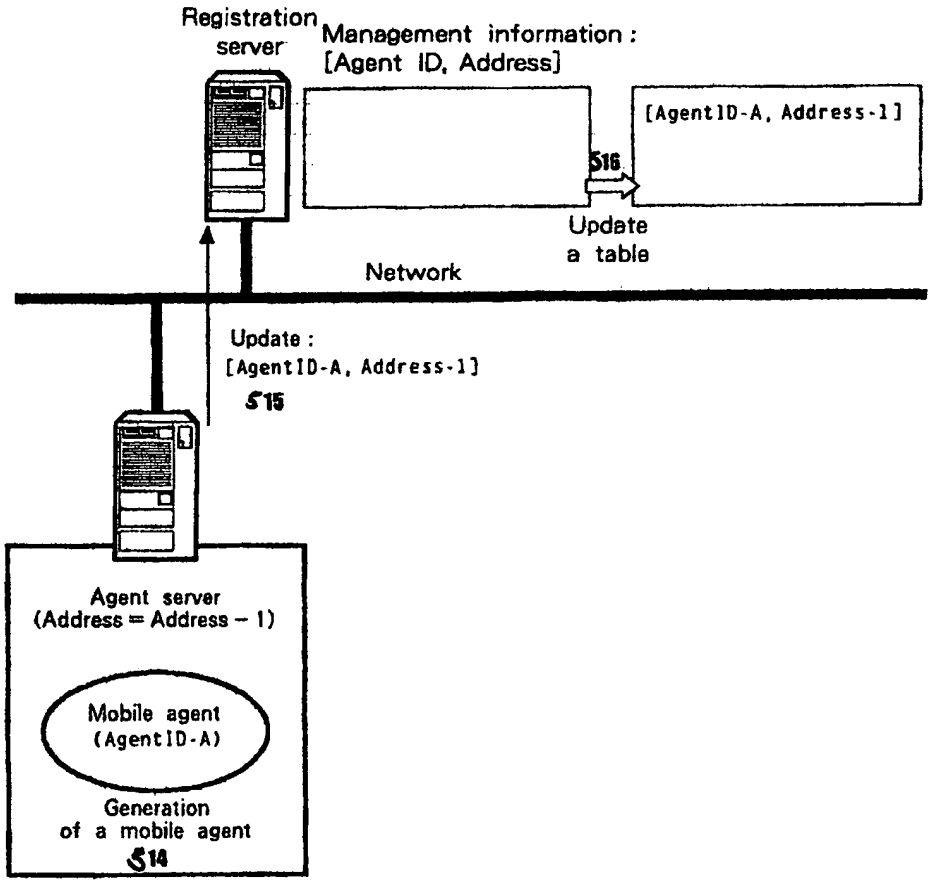
Registration server
Management information : [Agent ID, Address]
[AgentID-A, Address-1]
S16
Update a table
Network
Update : [AgentID-A, Address-1]
S15
Agent server (Address = Address – 1)
Mobile agent (AgentID-A)
Generation of a mobile agent
S14

Fig. 5

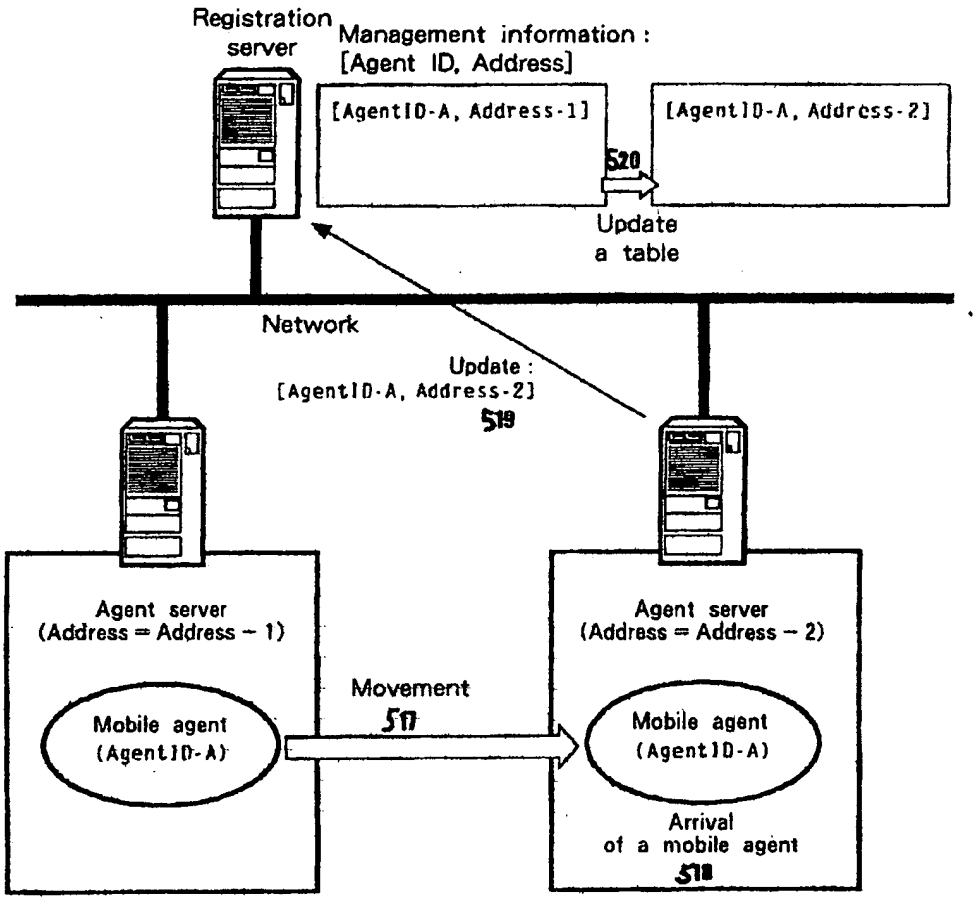
Registration server
Management information : [Agent ID, Address]
[AgentID-A, Address-1]
[AgentID-A, Address-2]
520
Update a table
Network
Update : [AgentID-A, Address-2]
519
Agent server (Address = Address – 1)
Agent server (Address = Address – 2)
Movement
517
Mobile agent (AgentID-A)
Mobile agent (AgentID-A)
Arrival of a mobile agent
518

Fig. 6

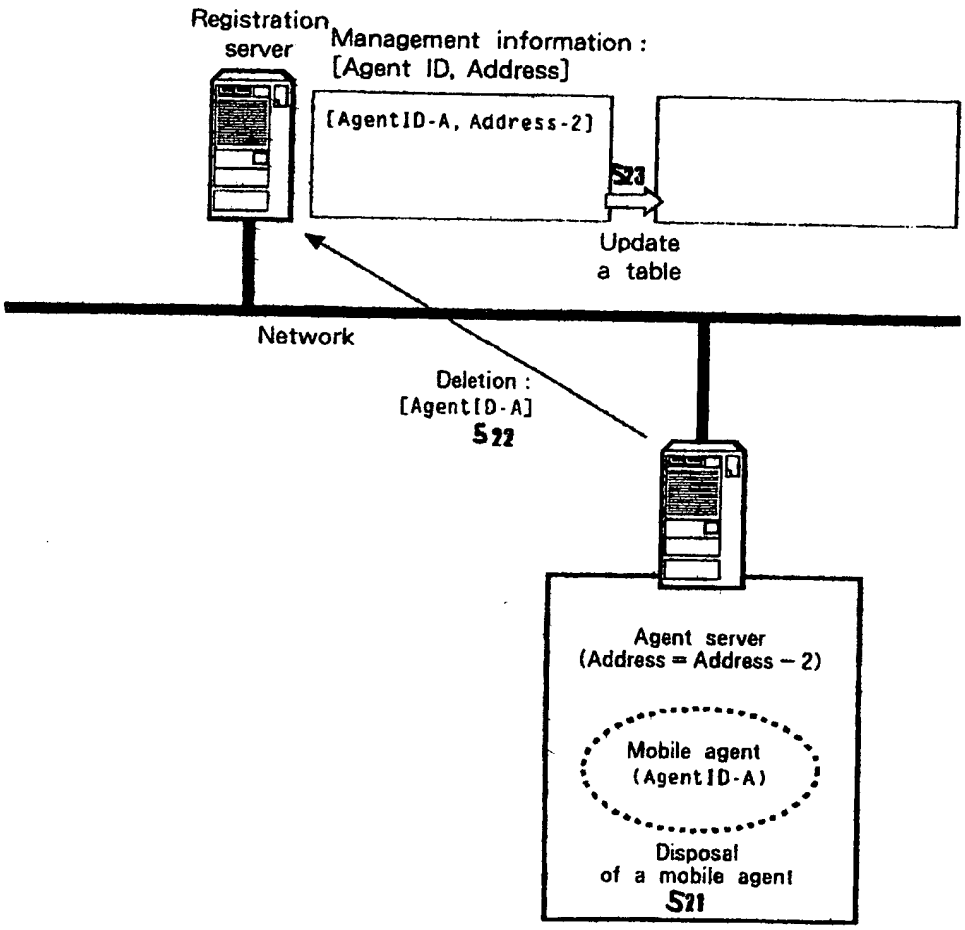
Registration server
Management information : [Agent ID, Address]
[AgentID-A, Address-2]
S23
Update a table
Network
Deletion : [AgentID-A]
S22
Agent server (Address = Address – 2)
Mobile agent (AgentID-A)
Disposal of a mobile agent
S21

Fig. 7

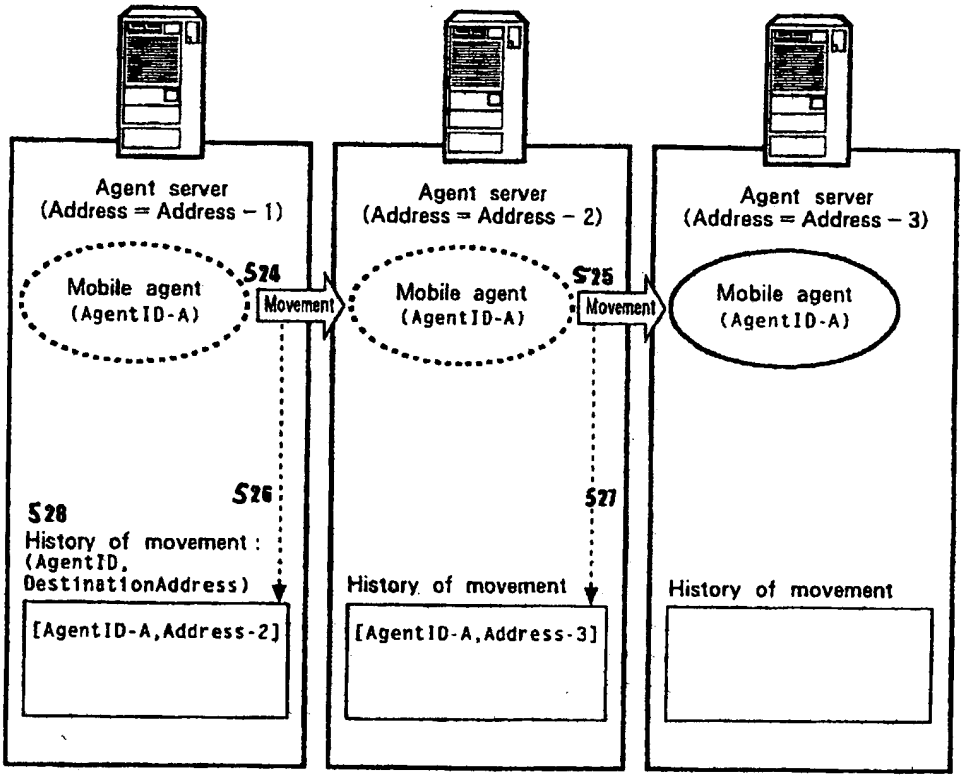
Agent server
(Address = Address – 1)
Mobile agent
(AgentID-A)
S24
Movement
S26
S28
History of movement :
(AgentID,
DestinationAddress)
[AgentID-A,Address-2]
Agent server
(Address = Address – 2)
Mobile agent
(AgentID-A)
S25
Movement
S27
History of movement
[AgentID-A,Address-3]
Agent server
(Address = Address – 3)
Mobile agent
(AgentID-A)
History of movement

Fig. 8

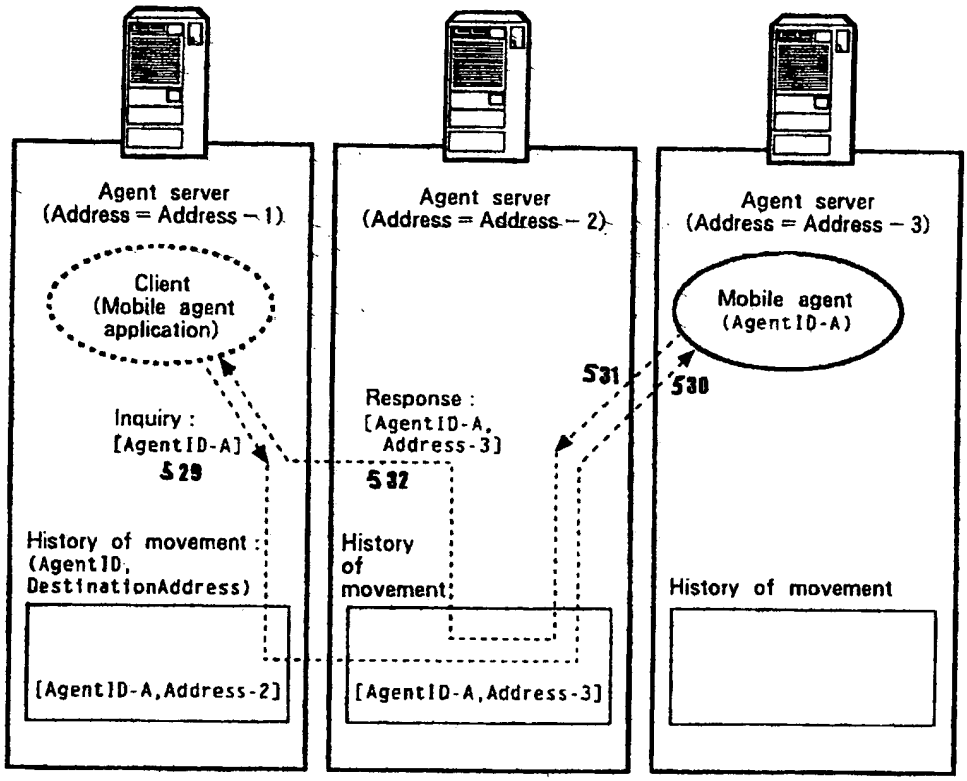
Agent server
(Address = Address – 1)
Client
(Mobile agent
application)
Inquiry :
[AgentID-A]
529
History of movement :
(AgentID,
DestinationAddress)
[AgentID-A,Address-2]
Agent server
(Address = Address – 2)
Response :
[AgentID-A,
Address-3]
532
History
of
movement
[AgentID-A,Address-3]
Agent server
(Address = Address – 3)
Mobile agent
(AgentID-A)
531
530
History of movement

Fig. 9

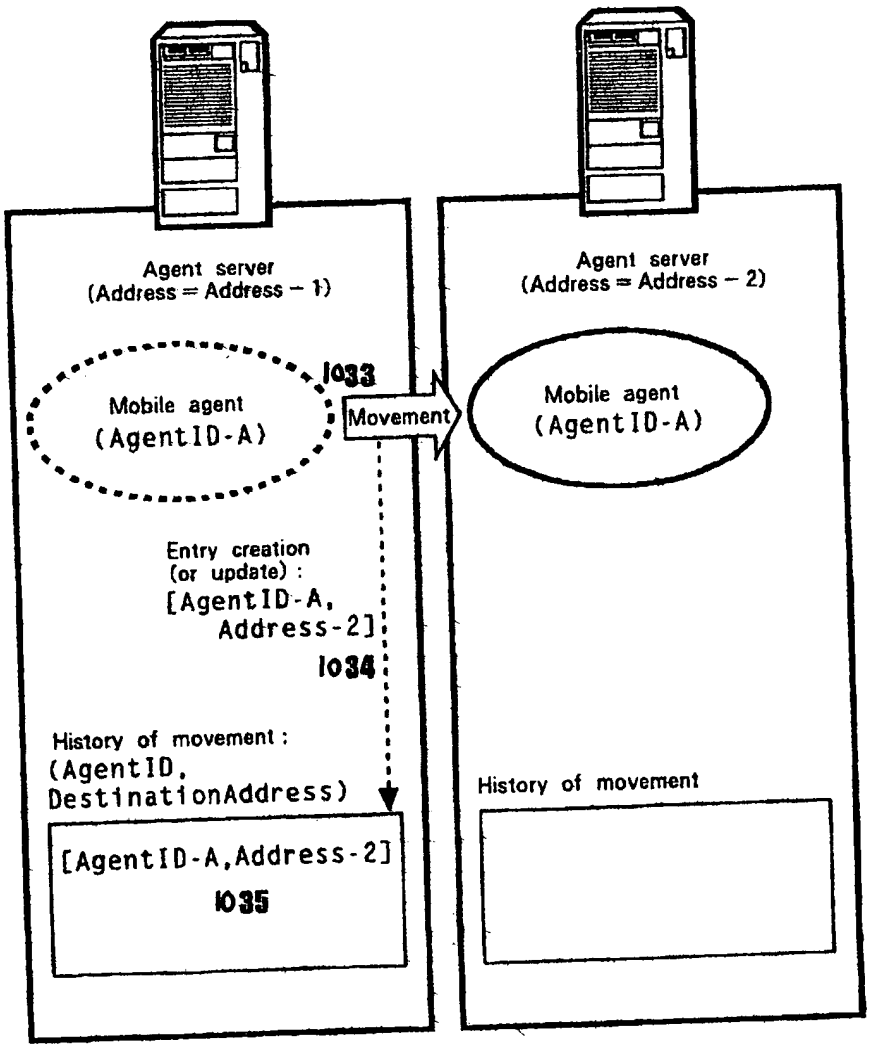
Agent server
(Address = Address – 1)
Agent server
(Address = Address – 2)
Mobile agent
(AgentID-A)
1033
Movement
Mobile agent
(AgentID-A)
Entry creation
(or update) :
[AgentID-A,
Address-2]
1034
History of movement :
(AgentID,
DestinationAddress)
[AgentID-A,Address-2]
1035
History of movement

Fig. 10

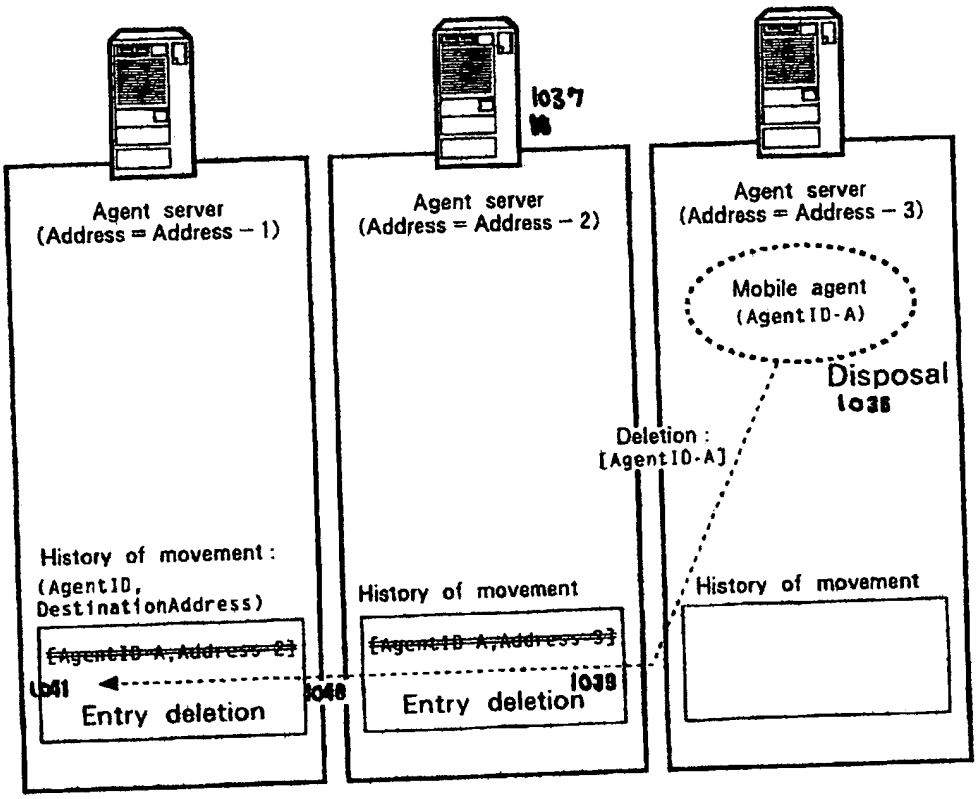
Agent server
(Address = Address – 1)
1037
Agent server
(Address = Address – 2)
Agent server
(Address = Address – 3)
Mobile agent
(AgentID-A)
Disposal
1038
Deletion :
[AgentID-A]
History of movement :
(AgentID,
DestinationAddress)
[AgentID-A,Address-2]
1041
Entry deletion
1040
History of movement
[AgentID-A,Address-3]
1039
Entry deletion
History of movement

Fig. 11

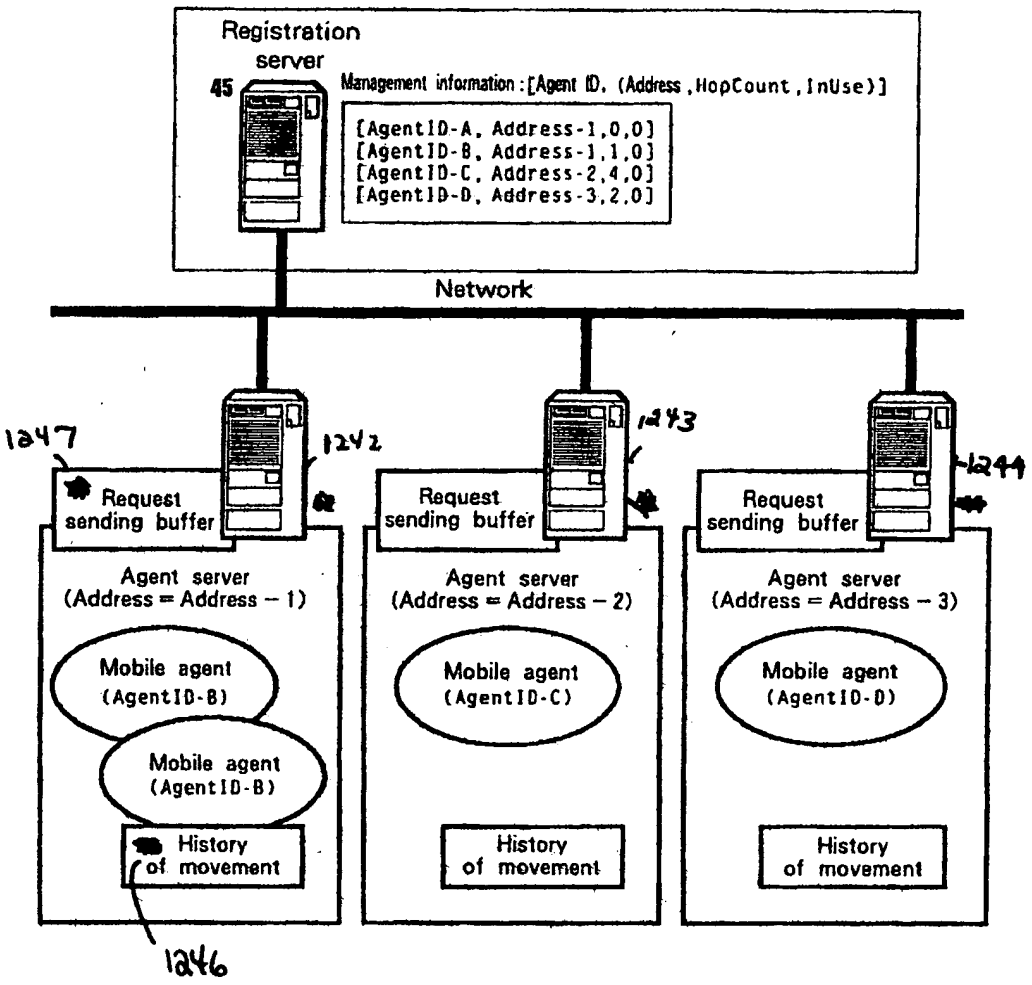
Registration server
45
Management information : [Agent ID, (Address, HopCount, InUse)]
[AgentID-A, Address-1,0,0]
[AgentID-B, Address-1,1,0]
[AgentID-C, Address-2,4,0]
[AgentID-D, Address-3,2,0]
Network
1247
1242
1243
1244
Request sending buffer
Request sending buffer
Request sending buffer
Agent server (Address = Address – 1)
Agent server (Address = Address – 2)
Agent server (Address = Address – 3)
Mobile agent (AgentID-B)
Mobile agent (AgentID-B)
Mobile agent (AgentID-C)
Mobile agent (AgentID-D)
History of movement
History of movement
History of movement
1246

Fig. 12

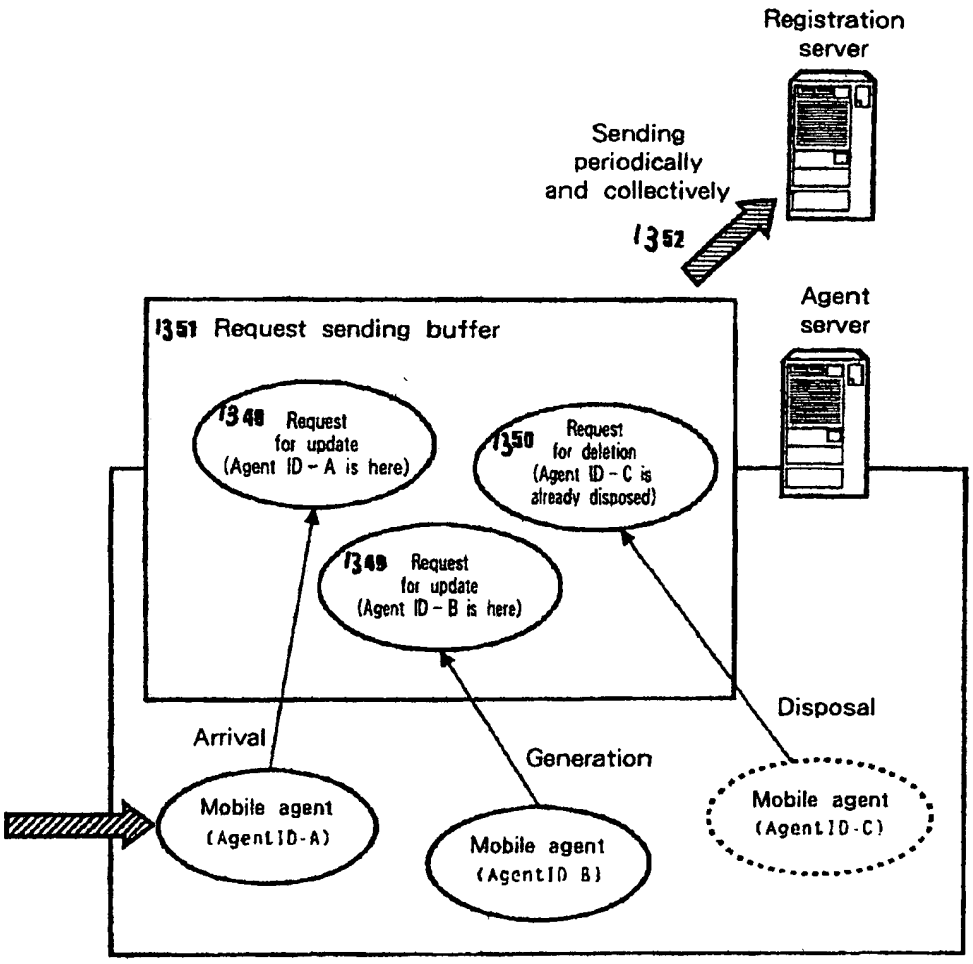
Registration server
Sending periodically and collectively
1352
Agent server
1351 Request sending buffer
1348 Request for update (Agent ID – A is here)
1350 Request for deletion (Agent ID – C is already disposed)
1349 Request for update (Agent ID – B is here)
Arrival
Generation
Disposal
Mobile agent (AgentID-A)
Mobile agent (AgentID B)
Mobile agent (AgentID-C)

Fig. 13

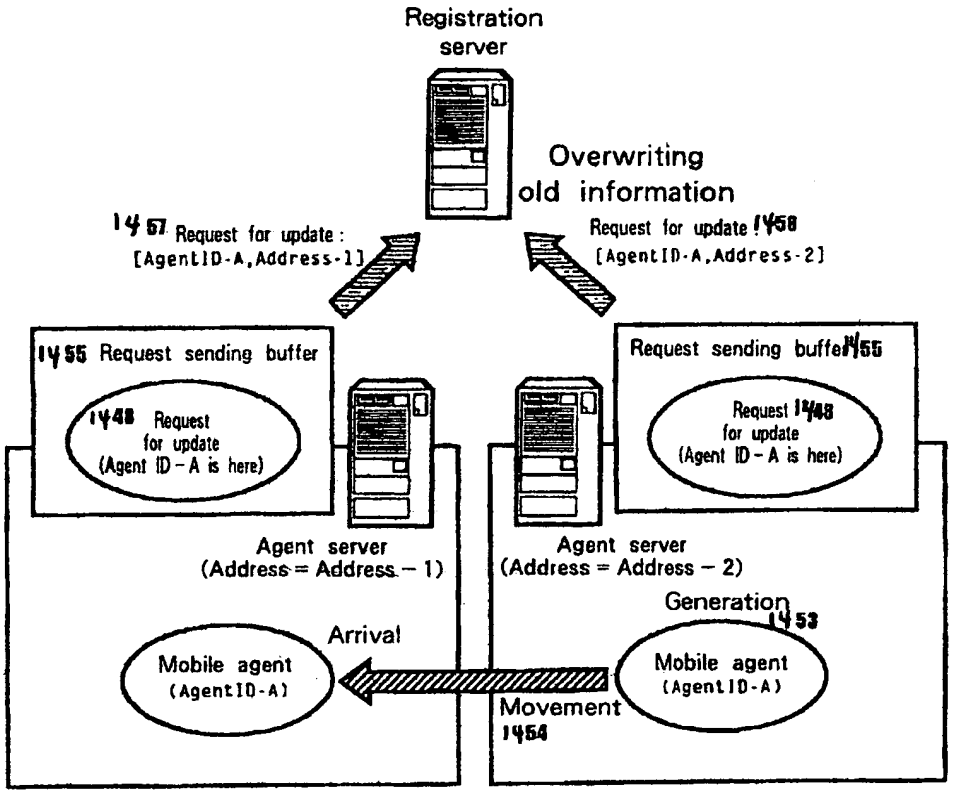
Registration server
Overwriting old information
1457 Request for update : [AgentID-A,Address-1]
Request for update 1458 [AgentID-A,Address-2]
1455 Request sending buffer
1448 Request for update (Agent ID – A is here)
Request sending buffer 1455
Request 1448 for update (Agent ID – A is here)
Agent server (Address = Address – 1)
Agent server (Address = Address – 2)
Generation 1453
Arrival
Mobile agent (AgentID-A)
Mobile agent (AgentID-A)
Movement 1454

Fig. 14

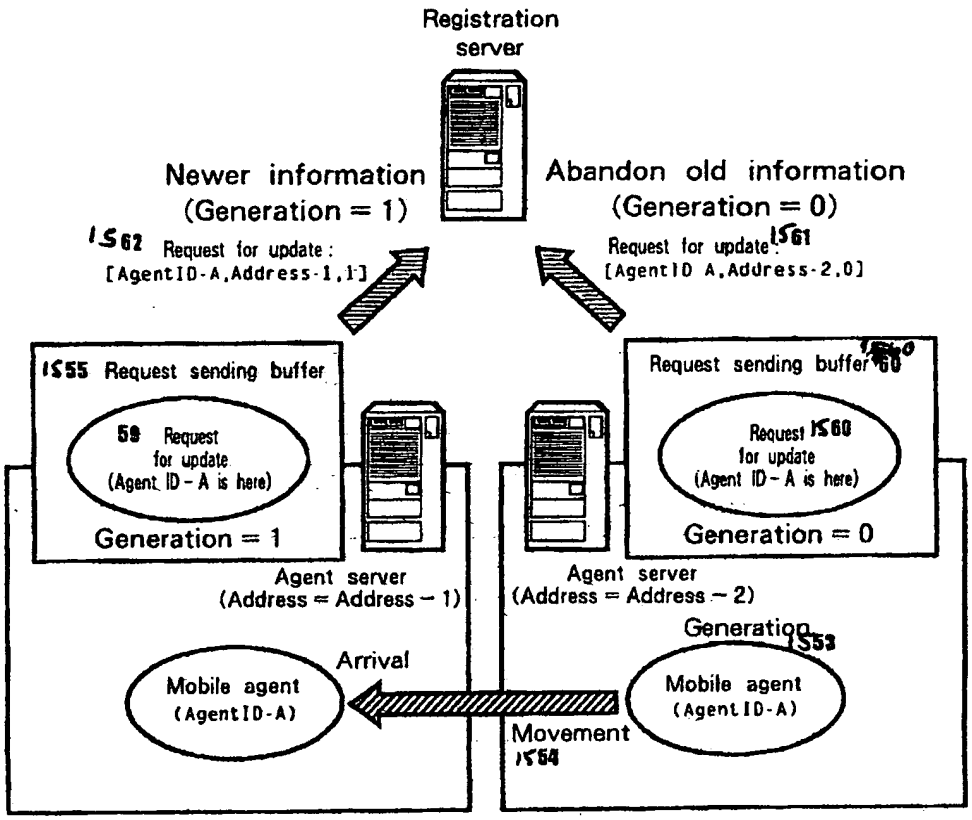
Registration server
Newer information (Generation = 1)
Abandon old information (Generation = 0)
1562 Request for update : [AgentID-A,Address-1,1]
1561 Request for update : [AgentID A,Address-2,0]
1555 Request sending buffer
59 Request for update (Agent ID-A is here)
Generation = 1
Agent server (Address = Address – 1)
Request sending buffer 1560
Request 1560 for update (Agent ID-A is here)
Generation = 0
Agent server (Address = Address – 2)
Generation 1553
Arrival
Mobile agent (AgentID-A)
Mobile agent (AgentID-A)
Movement 1554

Fig. 15

Requests to be stored in a request buffer

[AgentID, (Kind, HopCount)]

Agent
An ID of a subject mobile agent
Kind
A kind of a request
REGIST : Generation, on arrival (requests update of a management table of an arrival server)
UNREGIST : On disposal (requests deletion of an entry from a management table of a registration server)
HopCount
A request generation
When a request is generated, total accumulated number of movement of a mobile agent

Fig. 16

Entries of a management table of a registration server

[AgentID, (Address, HopCount,InUse)]

AgentID
An ID of a subject mobile agent
Address
An address of an agent server, which is a starting point of tracing a mobile agent
HopCount
A request generation
When a request is generated, total accumulated number of movement of a mobile agent
InUse
Entry use count
If 1 or more, it is in use
Used in order to prevent deletion of an entry or history of movement while tracing

Fig. 17

History of movement

[AgentID, Destination]

AgentID
An ID of a subject mobile agent
Address
An address of an agent server of a movement destination of a mobile agent

Fig. 18

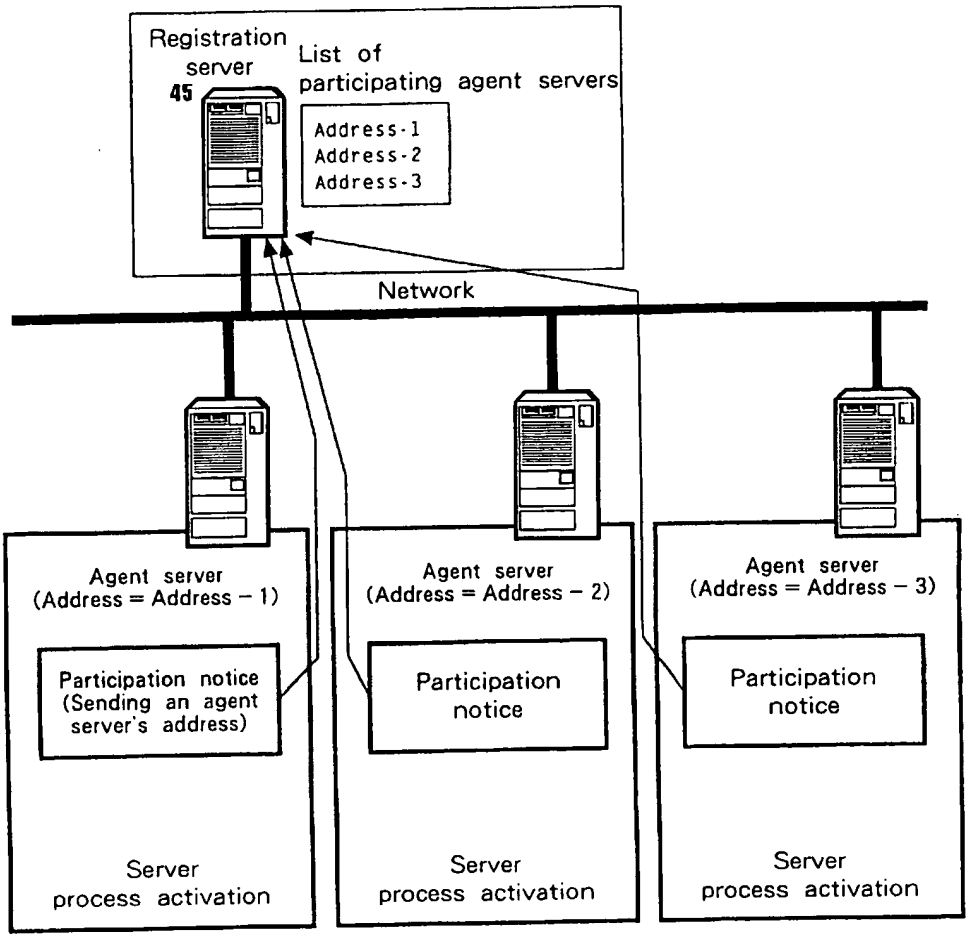
Registration server
45
List of participating agent servers
Address-1
Address-2
Address-3
Network
Agent server (Address = Address – 1)
Participation notice (Sending an agent server's address)
Server process activation
Agent server (Address = Address – 2)
Participation notice
Server process activation
Agent server (Address = Address – 3)
Participation notice
Server process activation

Fig. 19

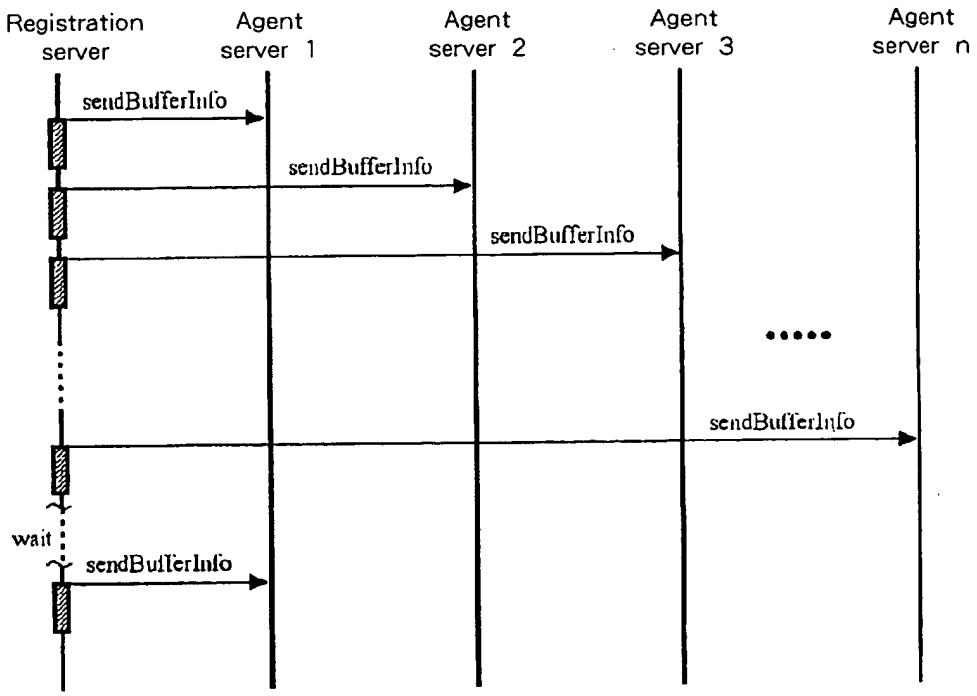
Registration server
Agent server 1
Agent server 2
Agent server 3
Agent server n
sendBufferInfo
sendBufferInfo
sendBufferInfo
sendBufferInfo
wait
sendBufferInfo

Fig. 20

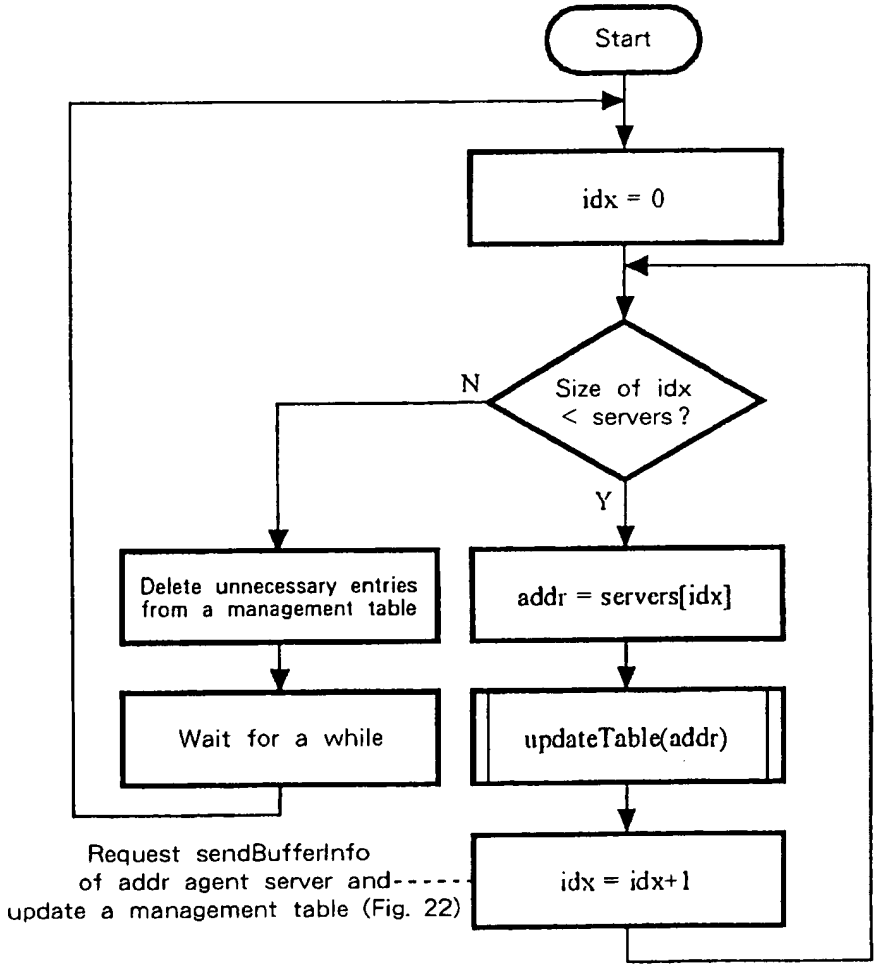
Start
idx = 0
Size of idx < servers ?
N
Y
Delete unnecessary entries from a management table
addr = servers[idx]
Wait for a while
updateTable(addr)
Request sendBufferInfo of addr agent server and update a management table (Fig. 22)
idx = idx+1

Fig. 21

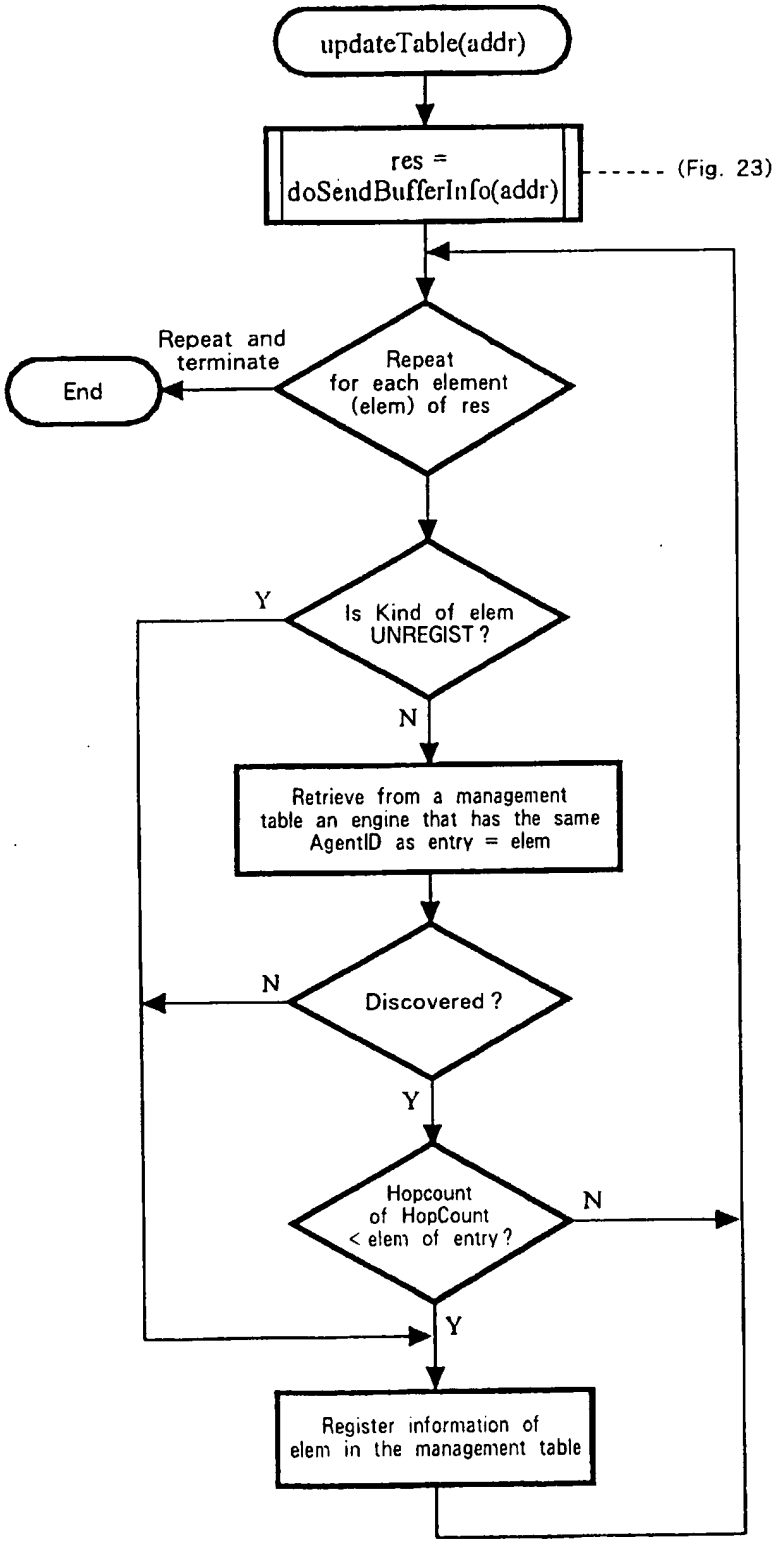
updateTable(addr)
res = doSendBufferInfo(addr)
(Fig. 23)
Repeat for each element (elem) of res
Repeat and terminate
End
Is Kind of elem UNREGIST ?
Y
N
Retrieve from a management table an engine that has the same AgentID as entry = elem
Discovered ?
N
Y
Hopcount of HopCount < elem of entry ?
N
Y
Register information of elem in the management table

Fig. 22

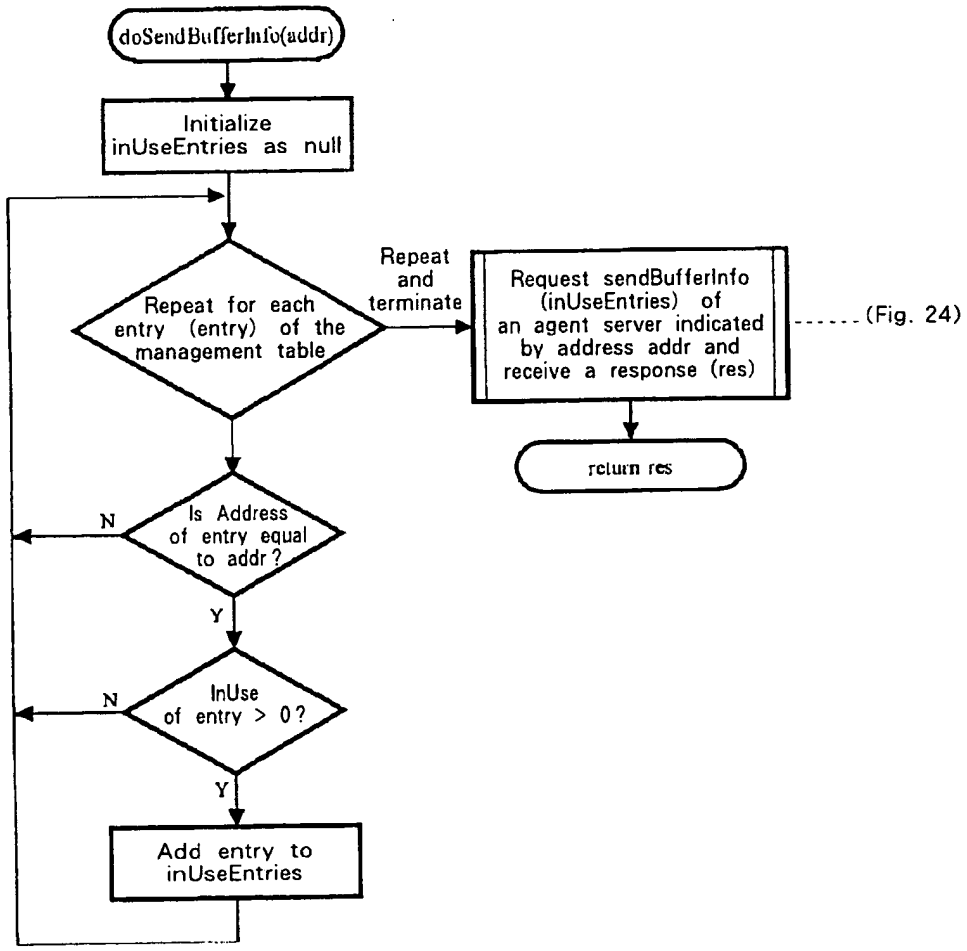
doSendBufferInfo(addr)
Initialize inUseEntries as null
Repeat for each entry (entry) of the management table
Repeat and terminate
Request sendBufferInfo (inUseEntries) of an agent server indicated by address addr and receive a response (res)
(Fig. 24)
return res
N
Is Address of entry equal to addr?
Y
N
InUse of entry > 0?
Y
Add entry to inUseEntries

Fig. 23

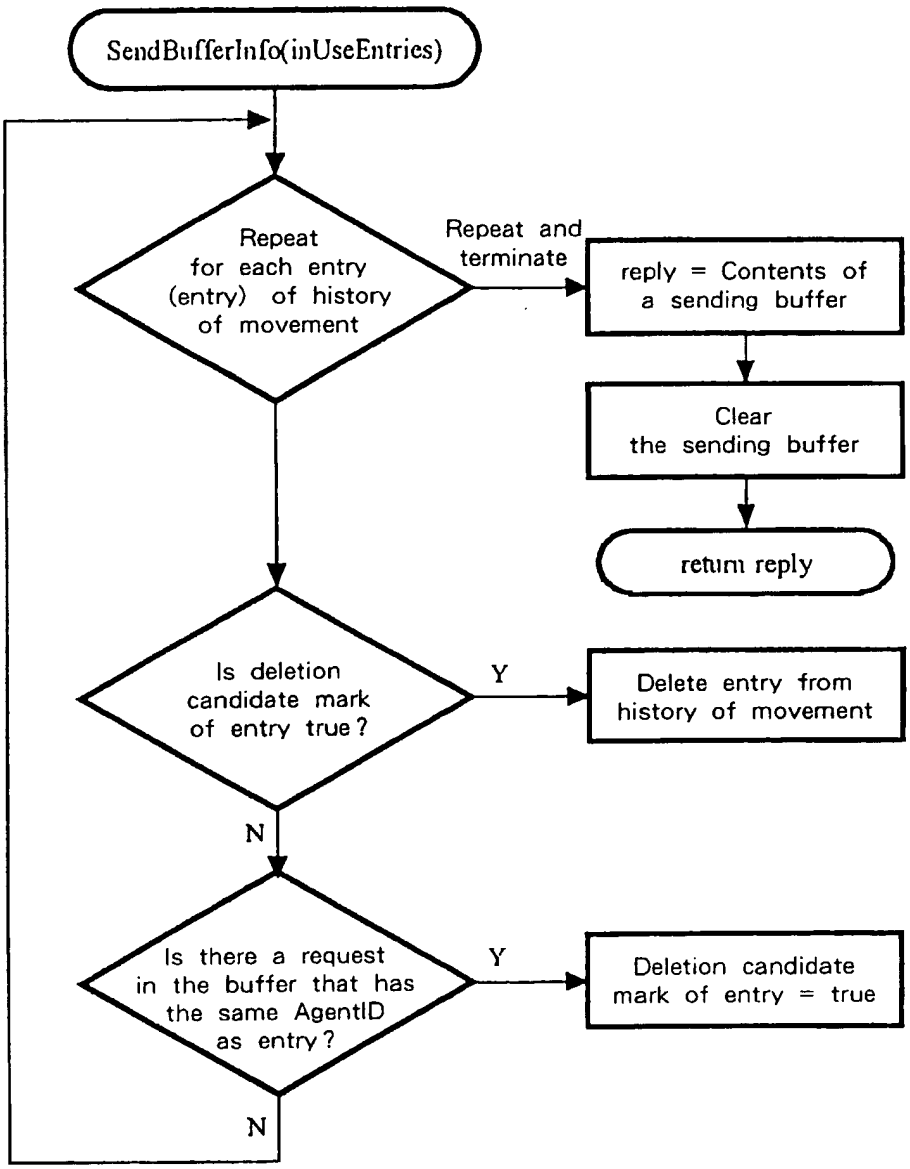
SendBufferInfo(inUseEntries)
Repeat for each entry (entry) of history of movement
Repeat and terminate
reply = Contents of a sending buffer
Clear the sending buffer
return reply
Is deletion candidate mark of entry true?
Y
Delete entry from history of movement
N
Is there a request in the buffer that has the same AgentID as entry?
Y
Deletion candidate mark of entry = true
N

Fig. 24

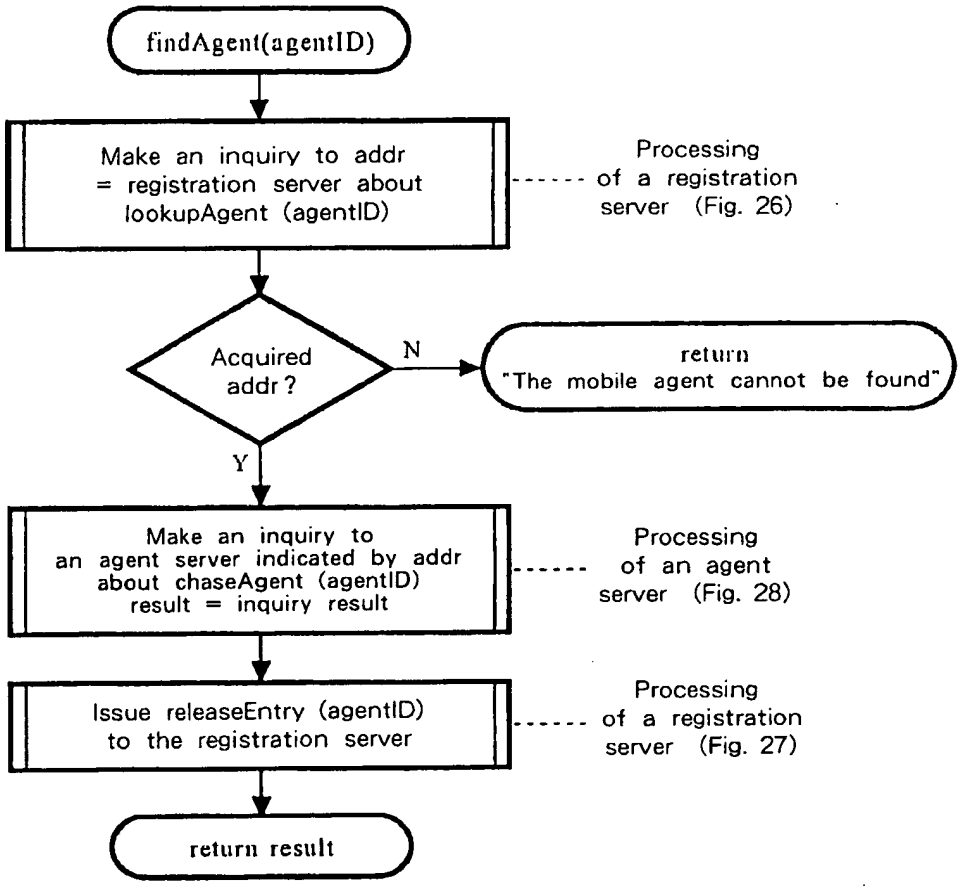
findAgent(agentID)
Make an inquiry to addr = registration server about lookupAgent (agentID)
Processing of a registration server (Fig. 26)
Acquired addr ?
N
return "The mobile agent cannot be found"
Y
Make an inquiry to an agent server indicated by addr about chaseAgent (agentID) result = inquiry result
Processing of an agent server (Fig. 28)
Issue releaseEntry (agentID) to the registration server
Processing of a registration server (Fig. 27)
return result

Fig. 25

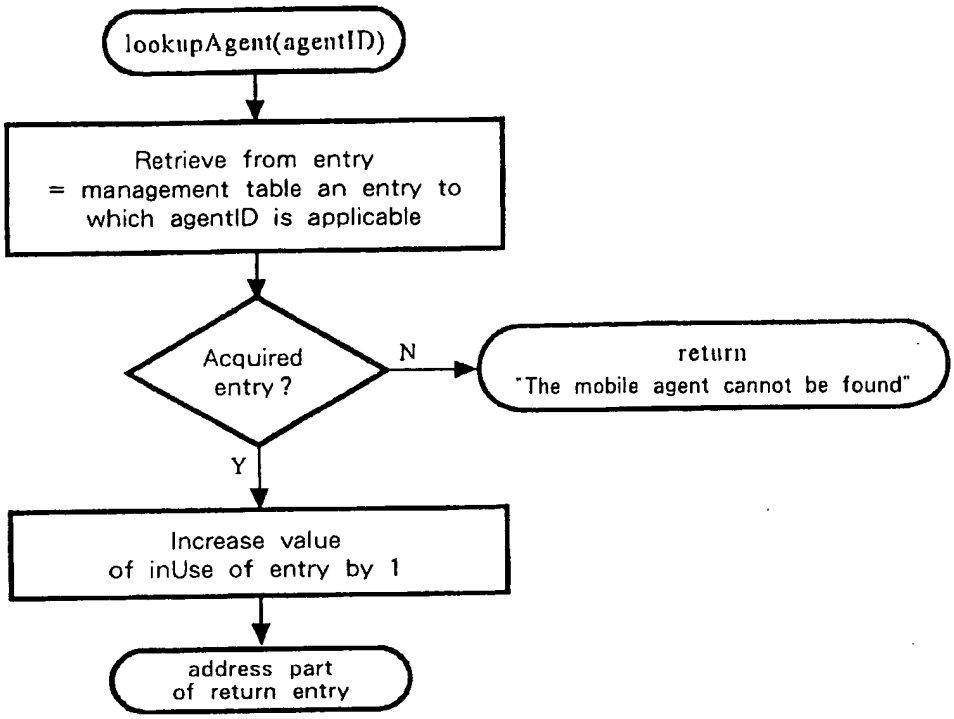
lookupAgent(agentID)
Retrieve from entry
= management table an entry to
which agentID is applicable
Acquired
entry ?
N
return
"The mobile agent cannot be found"
Y
Increase value
of inUse of entry by 1
address part
of return entry

Fig. 26

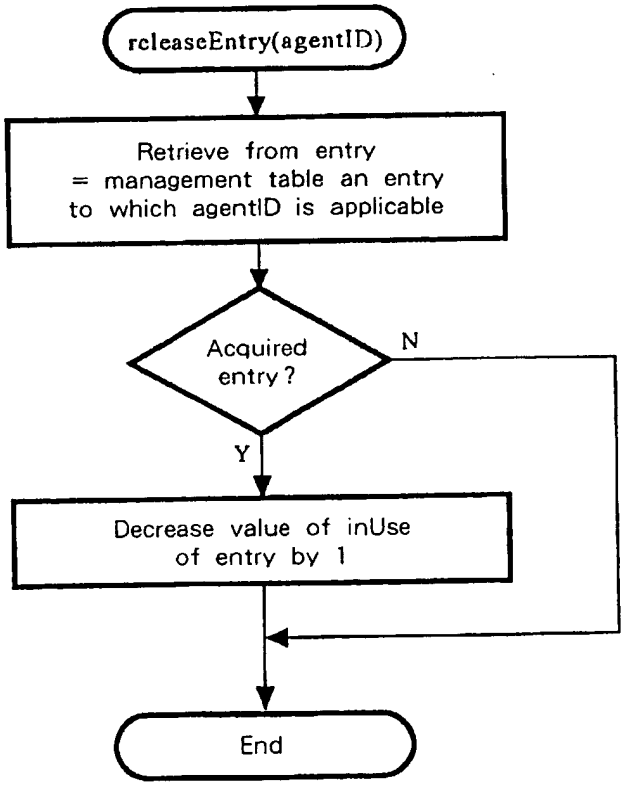
releaseEntry(agentID)
Retrieve from entry
= management table an entry
to which agentID is applicable
Acquired
entry ?
N
Y
Decrease value of inUse
of entry by 1
End

Fig. 27

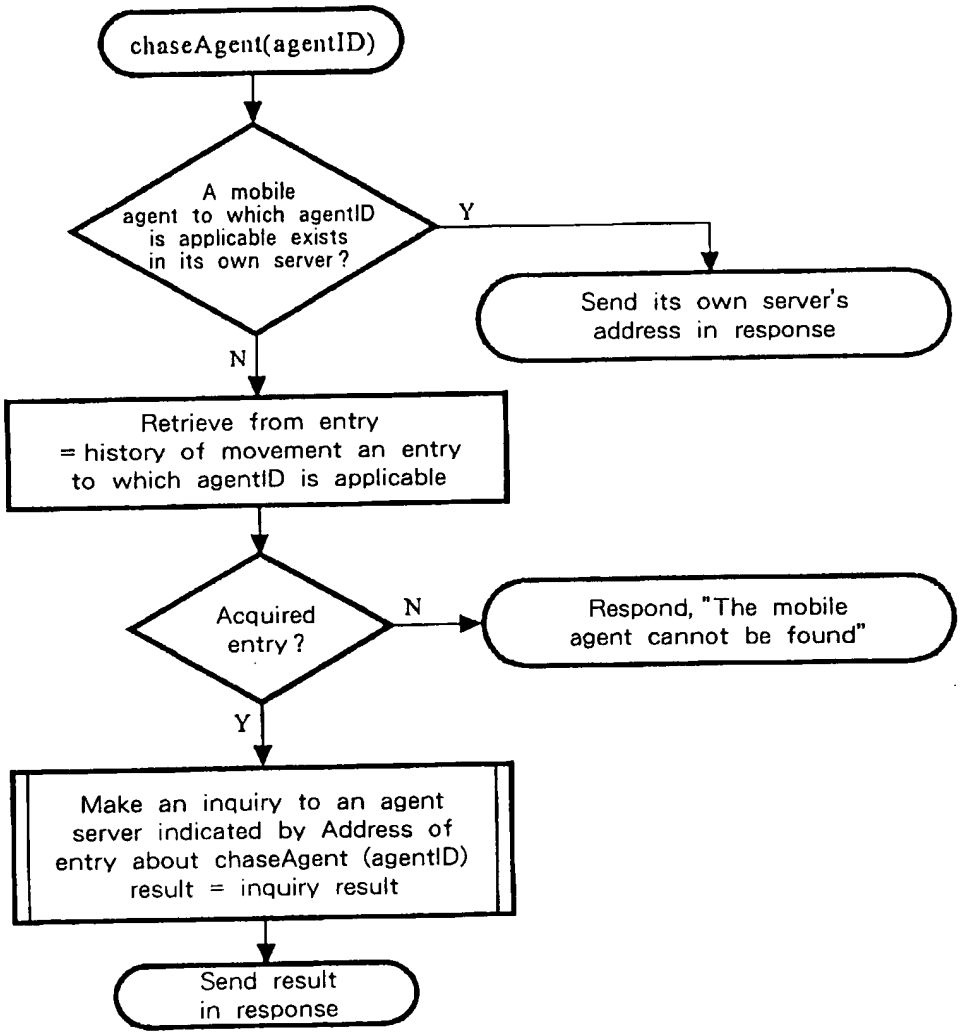
chaseAgent(agentID)
A mobile agent to which agentID is applicable exists in its own server?
Y
Send its own server's address in response
N
Retrieve from entry = history of movement an entry to which agentID is applicable
Acquired entry?
N
Respond, "The mobile agent cannot be found"
Y
Make an inquiry to an agent server indicated by Address of entry about chaseAgent (agentID) result = inquiry result
Send result in response

Fig. 28

Case 1 :
N = 9 (Total number of agents : 90)

Conventional method (Registry)	This method	No tracing facility
79,483	**63,599**	**61,084**
80,008	**66,079**	**65,241**
90,480	**66,079**	**66,504**

Fig. 29

Case 2 :
N = 15 (Total number of agents : 150)

Conventional method (Registry)	This method	No tracing facility
145,356	**100,240**	**97,002**
153,234	**107,596**	**108,289**
179,300	**109,654**	**112,066**

Fig. 30